

2/5/1 (Item 1 from file: 347)  
DIALOG(R) File 347:JAPIO  
(c) 2003 JPO & JAPIO. All rts. reserv.

02611314 \*\*Image available\*\*  
MULTIPLEX WINDOW CONTROL SYSTEM

PUB. NO.: 63-228214 A]  
PUBLISHED: September 22, 1988 (19880922)  
INVENTOR(s): TANI MASAYUKI  
YOKOYAMA TAKANORI  
ARAI TOSHIFUMI  
TANIFUJI SHINYA  
APPLICANT(s): HITACHI LTD [000510] (A Japanese Company or Corporation), JP  
(Japan)  
APPL. NO.: 62-060944 [JP 8760944]  
FILED: March 18, 1987 (19870318)  
INTL CLASS: [4] G06F-003/14  
JAPIO CLASS: 45.3 (INFORMATION PROCESSING -- Input Output Units)  
JOURNAL: Section: P, Section No. 816, Vol. 13, No. 30, Pg. 76, January  
24, 1989 (19890124)

#### ABSTRACT

PURPOSE: To improve operability, by detecting a window set in a specifically superposed state, and displaying a display object corresponding to a detected window.

CONSTITUTION: A window operation command execution part 4 interprets a bit of event information inputted from an input event managing part 3, and executes a window operation command based on a result. A hidden-surface detecting part 5 investigates the window being set as a hidden-surface currently, and as a result, when it exists, an icon managing part 6 is called, and an icon corresponding to a newly detected hidden-surface is displayed on a screen. In such a way, since it is possible to display the display object for the window by detecting the window set at the specifically superposed state and to operate the window for the display object directly, it can be evaded that the window fails to be operated or it is hard to be operated.

## ⑫ 公開特許公報(A)

昭63-228214

⑤Int.Cl.<sup>4</sup>

G 06 F 3/14

識別記号

3 5 0

庁内整理番号

7341-5B

⑬公開 昭和63年(1988)9月22日

審査請求 未請求 発明の数 1 (全10頁)

⑭発明の名称 多重ウィンドウ制御方式

⑮特 願 昭62-60944

⑯出 願 昭62(1987)3月18日

⑰発明者 谷 正 之 茨城県日立市久慈町4026番地 株式会社日立製作所日立研究所内  
 ⑱発明者 横 山 考 典 茨城県日立市久慈町4026番地 株式会社日立製作所日立研究所内  
 ⑲発明者 荒 井 俊 史 茨城県日立市久慈町4026番地 株式会社日立製作所日立研究所内  
 ⑳発明者 谷 藤 真 也 茨城県日立市久慈町4026番地 株式会社日立製作所日立研究所内  
 ㉑出 願 人 株式会社日立製作所 東京都千代田区神田駿河台4丁目6番地  
 ㉒代 理 人 弁理士 小川 勝男 外2名

## 明 細 書

## 1. 発明の名称

多重ウィンドウ制御方式

## 2. 特許請求の範囲

1. 多重ウィンドウシステムにおいて、特定の重り状態にあるウィンドウを検知する手段と、該検知手段によつて検知されたウィンドウに対応する表示物を表示する手段と、該表示手段によつて表示された表示物を指示することによつて該指示された表示物に対応するウィンドウに対する操作命令を実行せしめる手段とを備えたことを特徴とする多重ウィンドウ制御方式。
2. 該検知手段において、ウィンドウが特定の重り状態になった時点で特定の重り状態にあるウィンドウを検知することを特徴とする特許請求の範囲第1項記載の多重ウィンドウ制御方式。
3. 該検知手段において、ユーザから指示があった時点で特定の重り状態にあるウィンドウを検知することを特徴とする特許請求の範囲第1項記載の多重ウィンドウ制御方式。

4. 該検知手段において、他のウィンドウによつて完全に隠されたウィンドウのみを検知することを特徴とする特許請求の範囲第1～第3項の多重ウィンドウ制御方式。

5. 該操作命令実行手段において、該表示手段によつて表示された表示物が指示されたとき、該指示された表示物に対応するウィンドウをポップすることを特徴とする特許請求の範囲第1～4項の多重ウィンドウ制御方式。

## 3. 発明の詳細な説明

〔産業上の利用分野〕

本発明はワークステーションの多重ウィンドウ制御方式に係り、特に、ウィンドウの操作を容易にするのに好適な多重ウィンドウ制御方式に関する。

〔従来の技術〕

従来の多重ウィンドウシステムでは、例えばコンピュータ トウデイ (Computer Today), 1984/11, №4, 第23頁から第32頁に記載のように、CRT画面上に多くのウィンドウを重ねて

表示することができる。他のウインドウによつて一部分が隠されているウインドウAの全体を表示したい場合には、マウスと連動する画面上のカーソルをウインドウAの见えている部分に移動し、マウスのボタンを押下げるとウインドウA全体が一番上に表示される。このような、他のウインドウの下にあり一部又は全体が隠されているウインドウを一番上に表示し直す操作をポップと呼ぶ。

ウインドウA全体が他のウインドウによつて完全に隠されている場合には、カーソルをウインドウAの中におけないので、前記の方法ではウインドウAをポップすることができない。この場合には、上に表示されているウインドウを一番下にもつていく操作アンダーを用いる。すなわち、ウインドウAの一部分が見えるようになるまで、ウインドウAの上に重っているウインドウに上から順次アンダー操作を施し、ウインドウAの一部が見えたら、そこにカーソルを移動しマウスボタンを押すと、ウインドウAがポップできる。

〔発明が解決しようとする問題点〕

するウインドウに対する操作命令を実行する手段とを備えたものである。更に本発明においては、前記検知手段は、あるウインドウAが他のウインドウによつて完全に隠された時点で、ウインドウAを検知するように構成されており、更に前記操作命令実行手段は、前記指示された表示物に対応するウインドウをポップするように構成されていればさらによい。更に他、前記検知手段は、ユーザからの指示があつた時点で、他のウインドウによつて完全に隠されたウインドウを検知するように構成されていればさらによい。

〔作用〕

前記検知手段は特定の重なり状態にあるウインドウを検知し、そのウインドウ識別名を前記表示手段に報告する。報告を受けた表示手段は、報告のあつた識別名を有する各ウインドウに対応して識別可能を表示物を画面上の他のウインドウに隠されることのない領域に表示する。ユーザがこれらのうちの1つの表示物を指示すると、前記操作命令実行手段が指示された表示物に対応するウイ

上記従来技術においては、他のウインドウによつて完全に隠されてユーザから見えなくなつてしまつたウインドウの数が多くなつた場合が配慮されておらず、以下の問題があつた。

- (1) 見えているウインドウの下にどんなウインドウが隠れているかユーザがわからなくなる。
- (2) 完全に隠れてしまつたウインドウに対する操作を行うには、アンダー操作を試行錯誤的に用いて、上に重っているウインドウをどける必要があり、手間がかかる。

本発明の目的は、上記従来技術の問題点を解決し操作性の良い多重ウインドウ制御方式を実現することにある。

〔問題点を解決するための手段〕

上記目的を達成するため、本発明の多重ウインドウ制御方式は、特定の重なり状態にあるウインドウを検知する手段と、この検知手段によつて検知されたウインドウに対応する表示物を表示する手段と、この表示手段によつて表示された表示物を指示することによりこの指示された表示物に対応

ウインドウに対して特定の操作命令を実行する。

〔実施例〕

以下、本発明の一実施例を図に従つて説明する。

第1図は、本発明をワークステーションの多重ウインドウシステムに適用した場合の実施例を示すブロック図である。

第1図において、1はウインドウ操作コマンドを入力するためのキーボード、2は画面上のカーソルを連動し、付属のボタンを押下げ、解放することによりイベントを生成するマウス、3は、キーボード1やマウス2によつて生成されるイベント情報を順次とつてきて、そのイベント発生時点のカーソル位置に従つて他のブロックにイベント情報をわたす入力イベント管理部である。入力イベントには、キーボード1のキーの押下げ、マウス2に付属するボタンの押下げ、解放がある。各イベントが発生すると、入力管理部は、そのイベント発生時のカーソルの位置、イベントの種類、イベントを発生したデバイスの識別情報をひとまとめにしたイベント情報を生成する。更に、第1図

において、4は、入力イベント管理部3から入力されたイベント情報を解釈し、その結果に応じてウインドウ操作コマンドを実行するウインドウ操作実行部、5はウインドウ操作コマンドの実行により重なり状態が変化する可能性があるときに、ウインドウ操作コマンド実行後に他のウインドウにより完全に隠されてしまったウインドウ（このようなウインドウを隠面ウインドウと呼ぶ）の有無及び、隠面ウインドウの識別情報を検知する隠面検知部、6は、隠面検知部5によつて隠面ウインドウが検知された場合に、その隠面ウインドウの識別情報をアイコンとして表示するとともに、現在表示中のアイコンとそれに対応するウインドウとの対応表を管理し、更にユーザがアイコン指定したときにそのアイコンに対応する隠面ウインドウをポップするアイコン管理部である。

ウインドウ操作コマンド実行部4では以下のコマンドを実行する。

- (1) “生成”：新しいウインドウを生成する。新しく開かれたウインドウはすでに開かれている

のウインドウ管理情報は全て初期状態に戻される。

上記コマンドの指定は、キーボードからのコマンド名の入力、又はマウスによるメニューの選択により行う。第2図は、本実施例による画面構成を示している。第2図において、30は表示画面全体、31はウインドウ表示領域、32はアイコン表示領域、33は表示中のウインドウW1、34はウインドウW1によつて隠されている隠面ウインドウW1、35は隠面ウインドウW1に対応するアイコン、36はウインドウW2の識別名表示エリア、37～41はウインドウW1を操作するためのメニューであり、それぞれ前記コマンドの“移動”、“サイズ変更”、“アイコン化”、“アンダー”、“消去”対応し、42は、キーボードからのウインドウコマンド入力領域である。この第2図および第3図に示す本実施例の動作フロー図を用いて、第1図の各部の動作を説明する。

まず、ユーザがウインドウ操作メニューやアイコン上にカーソルを移動し、マウスボタンをクリック

ウインドウの上に表示される。

- (2) “消去”：表示中のウインドウを消去する。  
 (3) “サイズ変更”：表示中のウインドウの大きさを変更する。サイズ変更後のウインドウは表示中ウインドウの一番上に表示される。  
 (4) “移動”：表示中のウインドウの表示位置を変更する。移動後のウインドウは表示中のウインドウの一番上に表示される。  
 (5) “アイコン化”：表示中ウインドウの表示を消去し、そのウインドウの識別情報をもつアイコンを表示する。このアイコンをマウスボタンをクリックすると、もとのウインドウを再表示できる。ここでクリックとはマウスボタンを押し下げ後、解放する操作をいう。  
 (6) “アンダー”：指定したウインドウの表示優先度を最下位にして再表示する。  
 (7) “ポップ”：指定したウインドウの表示優先度を最上位にして再表示する。  
 (8) “終了”：ウインドウシステムを終了する。表示中の全ウインドウは消去され、システム内

クすると、入力イベント管理部1にユーザ操作に対応するイベント情報が入力される（ステップ11）。

イベント管理部は、イベント発生時点にカーソルがウインドウ領域31内にある場合には、ウインドウ操作実行部4に入力イベント情報をわたし、カーソルがアイコン領域32にある場合には、入力イベント情報をアイコン管理部6にわたす（ステップ12）。ウインドウ操作実行部4は、入力イベントがメニュー37～41上でのマウスボタンのクリックであれば、カーソル位置のメニュー項目に対応するウインドウ操作コマンドを実行し、キーボードからのコマンド入力であれば、カーソル位置のウインドウに対し、そのコマンドを実行し、メニュー以外のウインドウ内でのマウスボタンのクリックであればそのウインドウをポップし、それら以外のイベントであれば何もない（ステップ13）。ウインドウ操作実行部4は、実行したウインドウ操作コマンドがウインドウの重なり方に変化を及ぼす可能性がある場合には、隠面検知

部5を起動する(ステップ14)。すなわち、ウインドウ操作実行部4が“移動”、“サイズ変更”、“生成”、“消去”、“ポップ”、“アンダー”、“アイコン化”を実行した場合には、ウインドウの重なり方が変化し、新たな隠面ウインドウが生成されたり、隠面ウインドウであつたものが、一部表示可能になつたりするため、再度、隠面となつているウインドウを調べる必要がある。

隠面検知処理部では、現在隠面となつているウインドウを調べ(ステップ15)、その結果、隠面ウインドウがあれば(ステップ16)、アイコン管理部6を呼び出してアイコン表示処理を行い、画面上に新たに検知した隠面ウインドウに対応するアイコンを表示する。

一方、ステップ12において、入力イベント管理部はイベント発生時のカーソル位置がアイコン領域領域32内にある場合にはアイコン管理部に入力イベント情報をわたす。アイコン管理部は、わたされた入力イベントがアイコン領域32の表示中のアイコン上で発生したマウスボタンのクリ

ックであれば、そのアイコンに対応する隠面ウインドウを同定し(ステップ19)、隠面ウインドウをポップし(ステップ20)、その後、アイコン消去処理を行う(ステップ21)。

次に、第3図のステップ15の隠面検知処理について第4図～第11図を用いて詳しく説明する。第4図～第6図はウインドウ操作実行部14によつて設定されるウインドウ管理情報のうち隠面検知処理に必要な情報をテーブル化したものである。これらのテーブルは、隠面検知処理の最初に初期設定される。第4図(a)は現在画面上に表示中のウインドウN個それぞれの左上の頂点の座標( $x_1, y_1$ )と右下頂点の座標( $x_2, y_2$ )をウインドウ番号ごとに格納するウインドウ位置テーブルPos[i, j] ( $i=1, 2, \dots, N, j=1, 2, 3, 4$ )である。ウインドウ番号は、隠面検知部5が表示中のウインドウを識別するためにつけた番号である。同図(b)は、ウインドウ位置テーブルに格納してある座標値と画面上の座標系との対応を示している。

第5図は、ウインドウの表示優先度の高い順にウインドウ番号を格納した優先順位テーブルを表わす。Priority[1]に格納されているウインドウ番号のウインドウの表示優先度が最も高く、Priority[N]に格納されているウインドウ番号のウインドウの表示優先度が最も低い。

第6図は、隠面検知部5が便宜上つけたウインドウ番号と実際のウインドウに表示されるウインドウ名36との対応表である。Name[i]にはウインドウ番号iのウインドウ名が格納されている。

第7図は、隠面ウインドウを検知するための領域分割テーブルである。領域分割テーブルの各要素A[i, j]は、画面上の各ウインドウの境界線を延長したときにできる各矩形領域に対応しており、その矩形領域において最も表示優先度の高い(すなわち実際に表示される)ウインドウの番号がA[i, j]の値として格納される。従つて、領域分割テーブルの全ての要素に格納されていない番号のウインドウは画面上に全く表示されないウインドウ、すなわち隠面ウインドウである。第

8図は、画面上のウインドウと領域分割テーブルとの対応関係を示す。同図(a)は画面30上に2つのウインドウ1, 2が重ねて表示されている様子を示す。破線はウインドウ1, 2の境界線を延長したものであり、破線あるては実線によつて囲まれた矩形領域内のA[i, j]は、その矩形領域が領域分割テーブルの要素A[i, j]に対応していることを示している。一方、同図(b)は同図(a)のようにウインドウが表示されたときの、領域分割テーブルの内容を示しており、例えばA[2, 2]=2はA[2, 2]に対応する画面上の矩形領域(ウインドウ1と2とが重なりあつている領域)ではウインドウ2の内容が表示されていることを示す。

第9図は、各表示ウインドウが隠面ウインドウであるか否かを示すフラグを格納した隠面フラグテーブルFlog[i]の構成を示している。ここで、Flog[i]の値が0に等しければウインドウiは隠面ウインドウであることを示す。

次に、第10図の処理フロー図に従い、隠面検

知処理の手順を説明する。

ウインドウ操作実行部4から隠面検知部5に隠面検知処理が依頼されると、まずウインドウ操作実行部からウインドウ管理情報を入力し、第4図および第5図に示した一連のウインドウ管理テーブルの内容を設定する(ステップ101)。次に、各ウインドウの水平、垂直方向の境界線それぞれにその座標値の昇べきの順に1から通し番号をつける(ステップ102)。第11図に通し番号のつけ方の具体例を示す。同図では画面30上に2つのウインドウが重なって表示されており、水平、垂直それぞれ4本の境界線に、それぞれ上から1~4、及び左から1~4の通し番号がつけられている。

次に、領域分割テーブルA[i, j]の全要素を零に初期設定し(ステップ103)、全ウインドウに関して表示優先度の低い順に、各ウインドウの占める領域に対応する領域分割テーブルの要素に自分のウインドウ番号を格納していく。(ステップ104~107)表示優先度の低い順に領

iに対応するウインドウ番号を示す。Icon[i]の値が0の場合には、アイコンiに対応するウインドウがないこと、すなわちアイコンiが未使用であることを示す。

第13図は、アイコンiと画面30上の各アイコンの表示領域との対応関係を示している。アイコンは、アイコン表示領域32の上部から等間隔(図中L)に並べられる。図中Mはアイコンの最大数を表しており、Mは通常表示可能なウインドウの最大約-1に設定される。

第14図の処理フロー図に従い、アイコン表示処理(ステップ17)の詳細について説明する。

まず全てのウインドウに関して隠面フラグテーブルを探索し(ステップ140~141)、フラグ値が0だったらすでにアイコン管理テーブルに登録されているかを調べ(ステップ142)、登録されていない場合はアイコン管理テーブルを走査して未使用のアイコンを探し(ステップ143)、アイコン管理テーブルにウインドウ番号を登録した後、アイコン番号に対応する画面30上のア

域分割テーブルにウインドウ番号を書き込んでいくことにより、最終的に各領域においてもつとも表示優先度の高いウインドウ番号が領域分割テーブルに格納される。

次に、隠面フラグテーブルの全要素を零に初期設定した後(ステップ108)、領域分割テーブルの全要素を走査し(ステップ109)、領域分割テーブルウインドウ番号が格納されているか否かを調べ、格納されていればそのウインドウ番号に対応する隠面フラグテーブルの要素の値を1にする(ステップ110)以上の手順により、隠面フラグテーブルには、隠面ウインドウに対応する要素に0が、その他のところには1が格納されることになる。

次に、アイコン表示処理(第3図ステップ17)の詳細について第12図~第15図を用いて説明する。

第12図は、アイコンとウインドウ番号との対応を管理するアイコン管理テーブルIcon[i]の構成を示している。Icon[i]は、アイコン番号

アイコン表示領域に、ウインドウ名を表示する(ステップ145)。ウインドウ名は、ウインドウ名テーブル(第6図)により知ることができる。

一方、ステップ141においてフラグ値が1の場合は、ウインドウWnoは隠面ウインドウではないので、以前に隠面ウインドウとしてアイコン管理テーブルに登録されているかを調べ(ステップ146)、もし登録されていればそれを消去する。

本実施例においては、ウインドウの重なり状態が変化するごとに隠面検知処理を実行し、常時隠面ウインドウに対応するウインドウを表示していたが、ユーザが指定したときだけ、隠面検知処理を行いその結果をアイコン表示し、ユーザがアイコンを選択した後、あるいは他のウインドウ操作を指定した後はアイコン表示を消去するようにしてもよい。これにより、処理の効率と画面スペースの利用効率を高めることができる。

さらに、本実施例ではアイコン選択時にはそのアイコンに対応する隠面ウインドウのポップ処理のみを行ったが、アイコン選択時に前述した各種

ウィンドウ操作コマンドをメニューとして表示し、各種のウィンドウ操作を隠面ウィンドウに直接施せるようにしてもよい。こうすることにより、隠面ウィンドウも他のウィンドウと同様直接操作できるようになり使い勝手が向上する。

〔発明の効果〕

本発明によれば、特定の重なり状態にあるウィンドウを検知して、そのウィンドウに対する表示物を表示し、その表示物を選択するとその表示物に対応するウィンドウを直接操作できるため、ウィンドウの重なり状態によつて、ウィンドウを操作できなくなったり、操作しにくくなることがない。さらに、ウィンドウの重り状態によらず、ウィンドウに関する一定以上の情報をユーザに提供できる。

4. 画面の簡単な説明

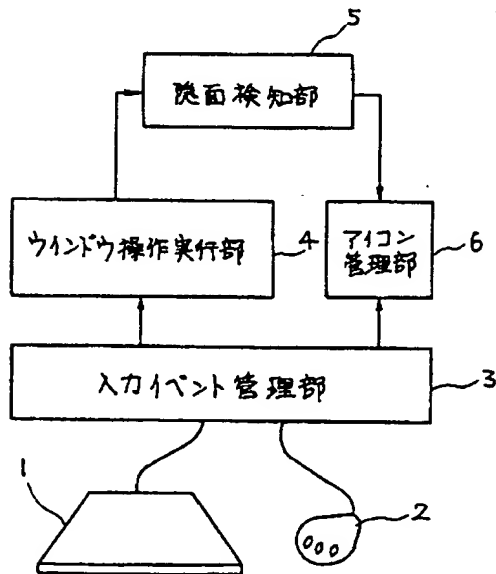
第1図は本発明の一実施例を示すブロック図、第2図は画面構成図、第3図は第1図の動作フロー図、第4図はウィンドウ位置テーブル構成図、第5図は優先順位テーブル構成図、第6図はウィ

ンドウ名テーブル構成図、第7図は領域分割テーブル構成図、第8図は領域分割テーブルの具体例を示す図、第9図は隠面フラグテーブル構成図、第10図は隠面検知処理のフロー図、第11図は境界線番号付けの具体例を示す図、第12図はアイコン管理テーブル構成例を示す図、第13図はアイコン表示領域構成図、第14図はアイコン表示処理のフロー図である。

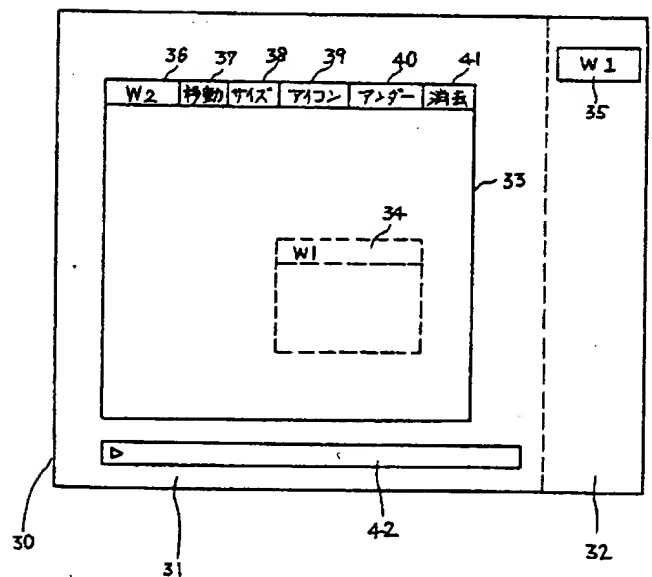
1…キーボード、2…マウス、3…入力イベント管理部、4…ウィンドウ操作管理部、5…隠面検知部、6…アイコン管理部、30…画面、31…ウィンドウ表示領域、32…アイコン表示領域。

代理人 井理士 小川勝男

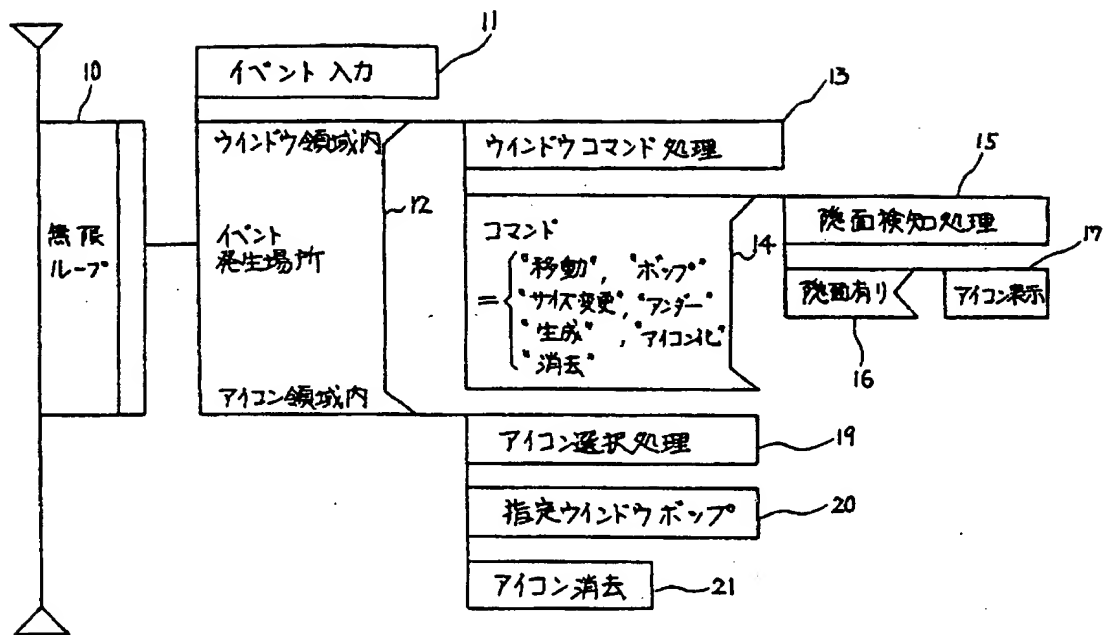
第 1 図



第 2 図

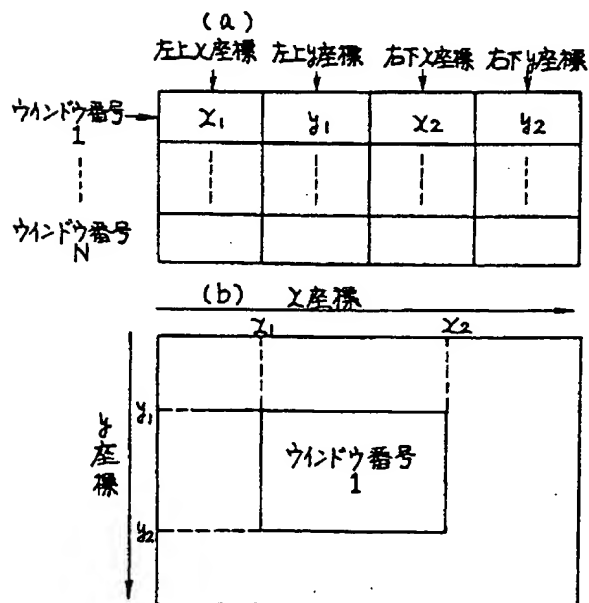


第 3 図



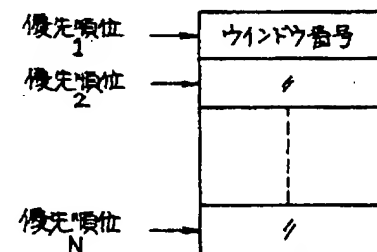
第 4 図

ウィンドウ位置テーブル Pos [i, j]



第 5 図

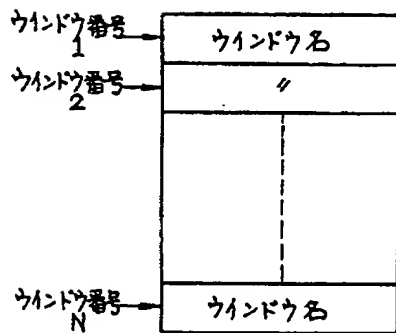
優先順位テーブル Priority [i]





第 6 図

ウィンドウ名テーブル Name [i]



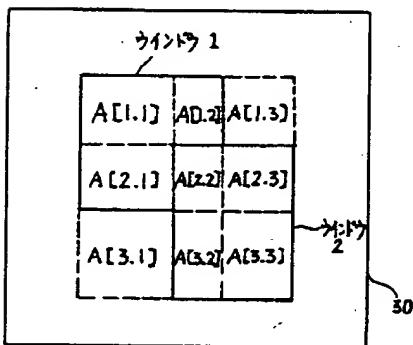
第 7 図

領域分割テーブル A[i, j]

	1	2	-----	2N-1
1	ウィンドウ 番号	◇	-----	◇
2	◇	◇	-----	◇
i	◇	◇	◇	◇
2N-1	◇	◇	-----	◇

第 8 図

(a)

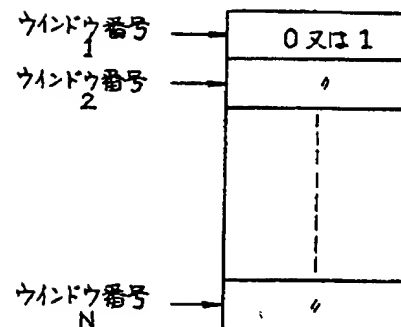


(b)

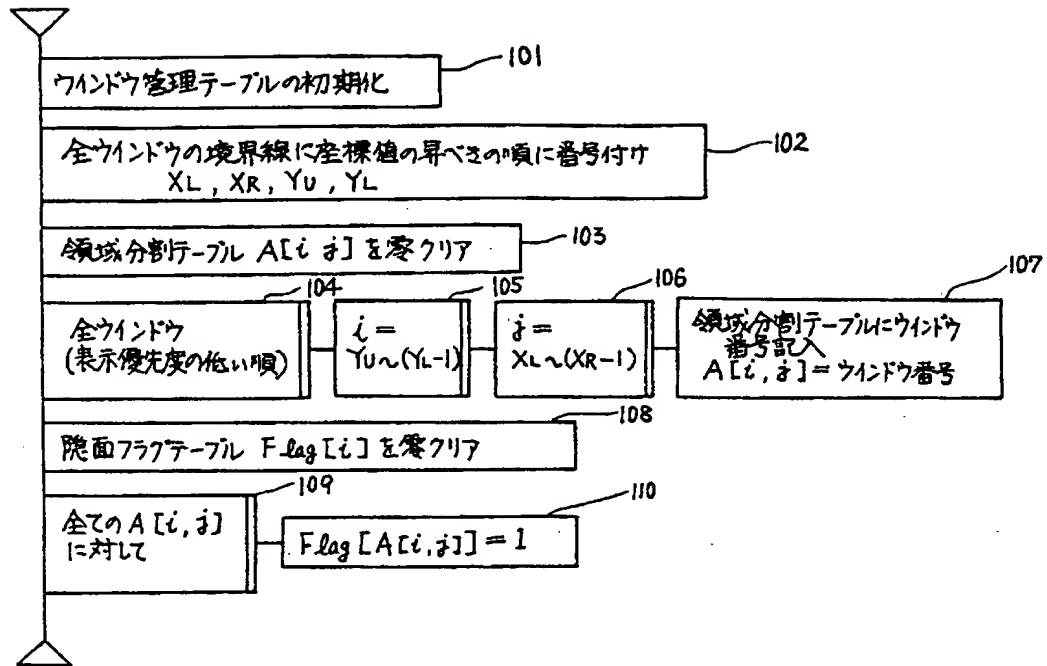
A[1.1] = 1	A[1.2] = 1	A[1.3] = ◇
A[2.1] = 1	A[2.2] = 2	A[2.3] = 2
A[3.1] = 0	A[3.2] = 2	A[3.3] = 2

第 9 図

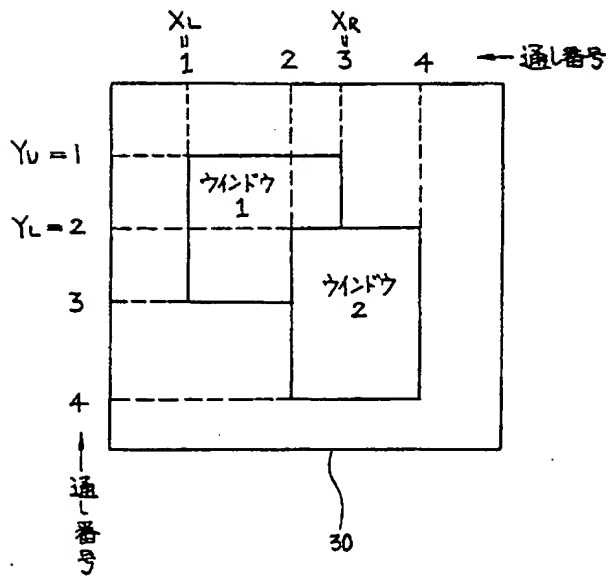
隣面フラグテーブル Flag [i]



第 10 図

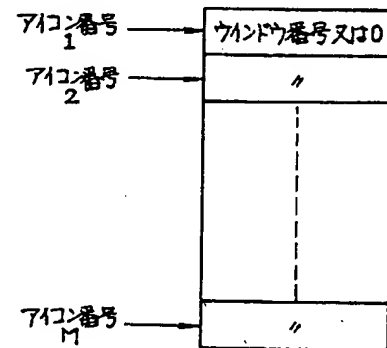


第 11 図

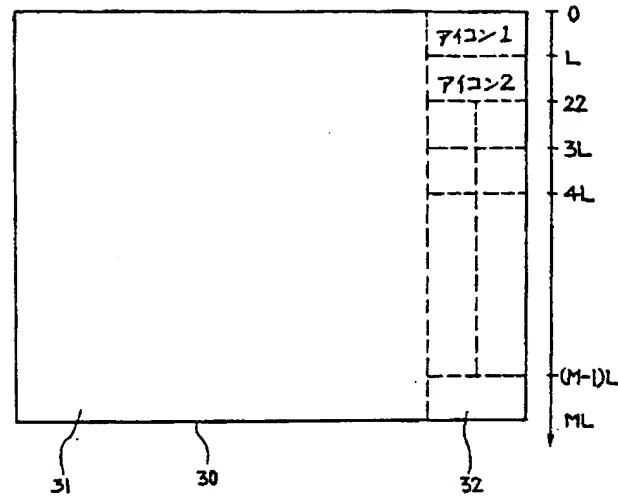


第 12 図

アイコン管理テーブル Icon[i]



第 13 図



第 14 図

